Signal Processing

Young Won Lim
1/19/18

Please send corrections (or suggestions) to youngwlim@hotmail.com.

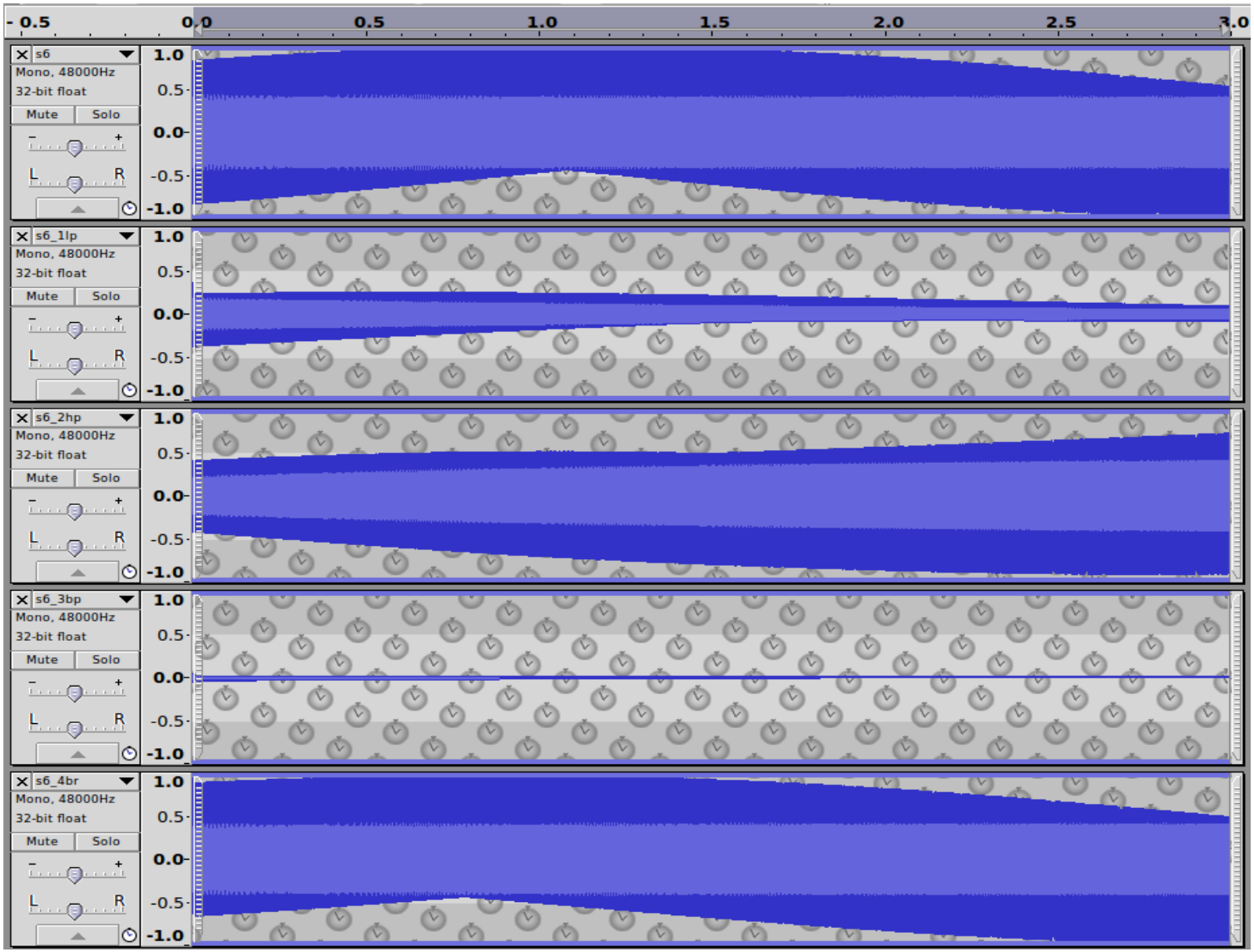
This document was produced by using LibreOffice.

Based on

Signal Processing with Free Software : Practical Experiments
F. Auger

```
sox s6.wav s6_1lp.wav  lowpass    -1 500
sox s6.wav s6_2hp.wav highpass   -2 2000 0.8q
sox s6.wav s6_3bp.wav bandpass  200 5q
sox s6.wav s6_4br.wav  bandreject   200 0.6q
sox s6.wav s6_5bp.wav  gain -20
```

Generating signals using sox



--plot gnuplot | octave

```
sox --plot gnuplot s6.wav -n lowpass    -1 500          > 1.plt
sox --plot gnuplot s6.wav -n highpass   -2 2000 0.8q    > 2.plt
sox --plot gnuplot s6.wav -n bandpass   200 5q          > 3.plt
sox --plot gnuplot s6.wav -n bandreject   200 0.6q      > 4.plt
```

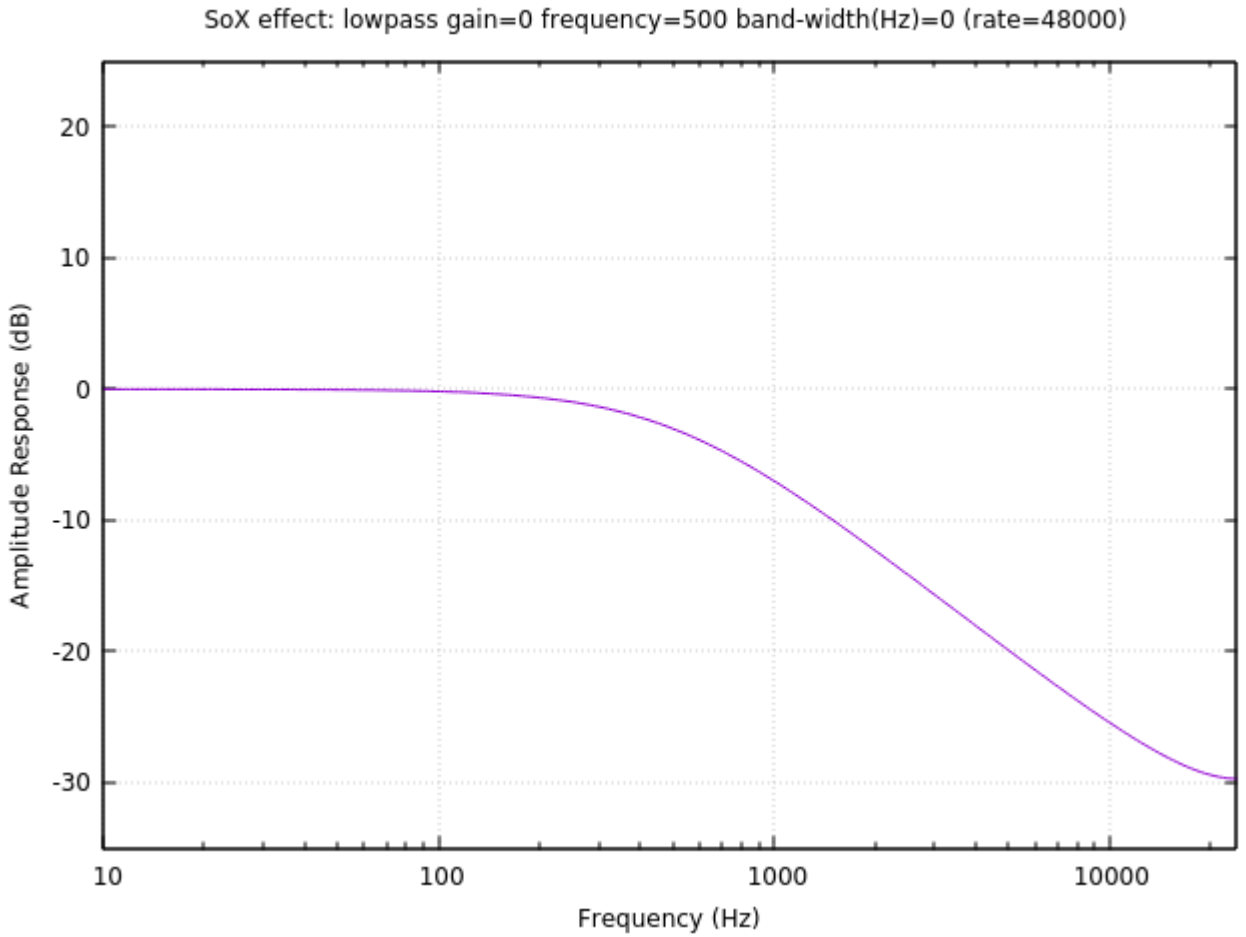
Gnuplot file

```
# gnuplot file
set title 'SoX effect: lowpass gain=0 frequency=500 band-width(Hz)=0
(rate=48000
)'
set xlabel 'Frequency (Hz)'
set ylabel 'Amplitude Response (dB)'
Fs=48000
b0=6.335397876340410e-02; b1=0.000000000000000e+00;
b2=0.000000000000000e+00; a1
=-9.366460212365959e-01; a2=0.000000000000000e+00
o=2*pi/Fs
H(f)=sqrt((b0*b0+b1*b1+b2*b2+2.*(b0*b1+b1*b2)*cos(f*o)
+2.*(b0*b2)*cos(2.*f*o))/(
1.+a1*a1+a2*a2+2.*(a1+a1*a2)*cos(f*o)+2.*a2*cos(2.*f*o)))
set logscale x
set samples 250
set grid xtics ytics
set key off
plot [f=10:Fs/2] [-35:25] 20*log10(H(f))
pause -1 'Hit return to continue'
```

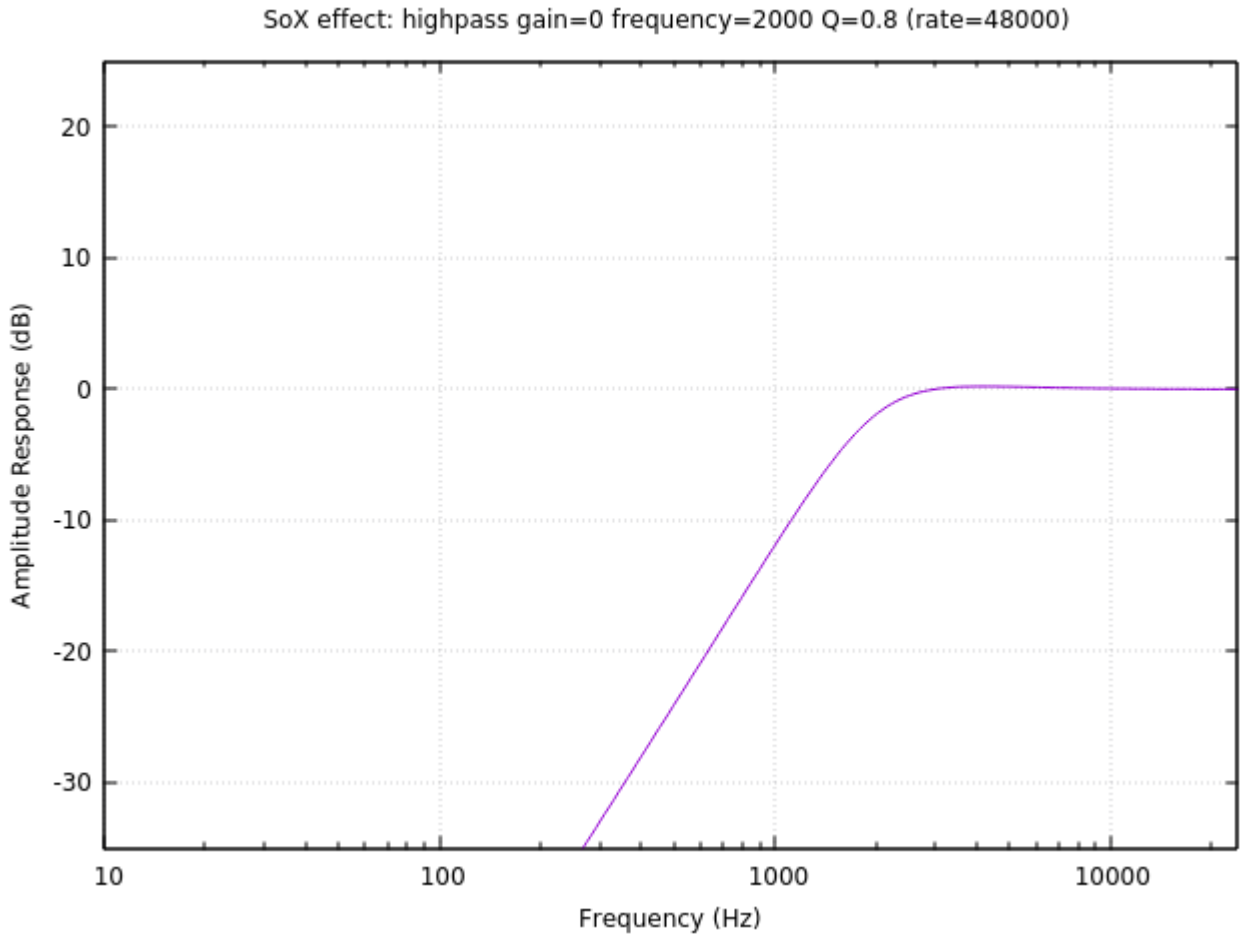
Octave file

```
% GNU Octave file (may also work with MATLAB(R) )
Fs=48000;minF=10;maxF=Fs/2;
sweepF=logspace(log10(minF),log10(maxF),200);
[h,w]=freqz([6.33539787634041e-02 0.00000000000000e+00
0.00000000000000e+00],
[1 -9.36646021236595e-01 0.00000000000000e+00],sweepF,Fs);
semilogx(w,20*log10(h))
title('SoX effect: lowpass gain=0 frequency=500 band-width(Hz)=0 (rate=48000)')
xlabel('Frequency (Hz)')
ylabel('Amplitude Response (dB)')
axis([minF maxF -35 25])
grid on
disp('Hit return to continue')
pause
```

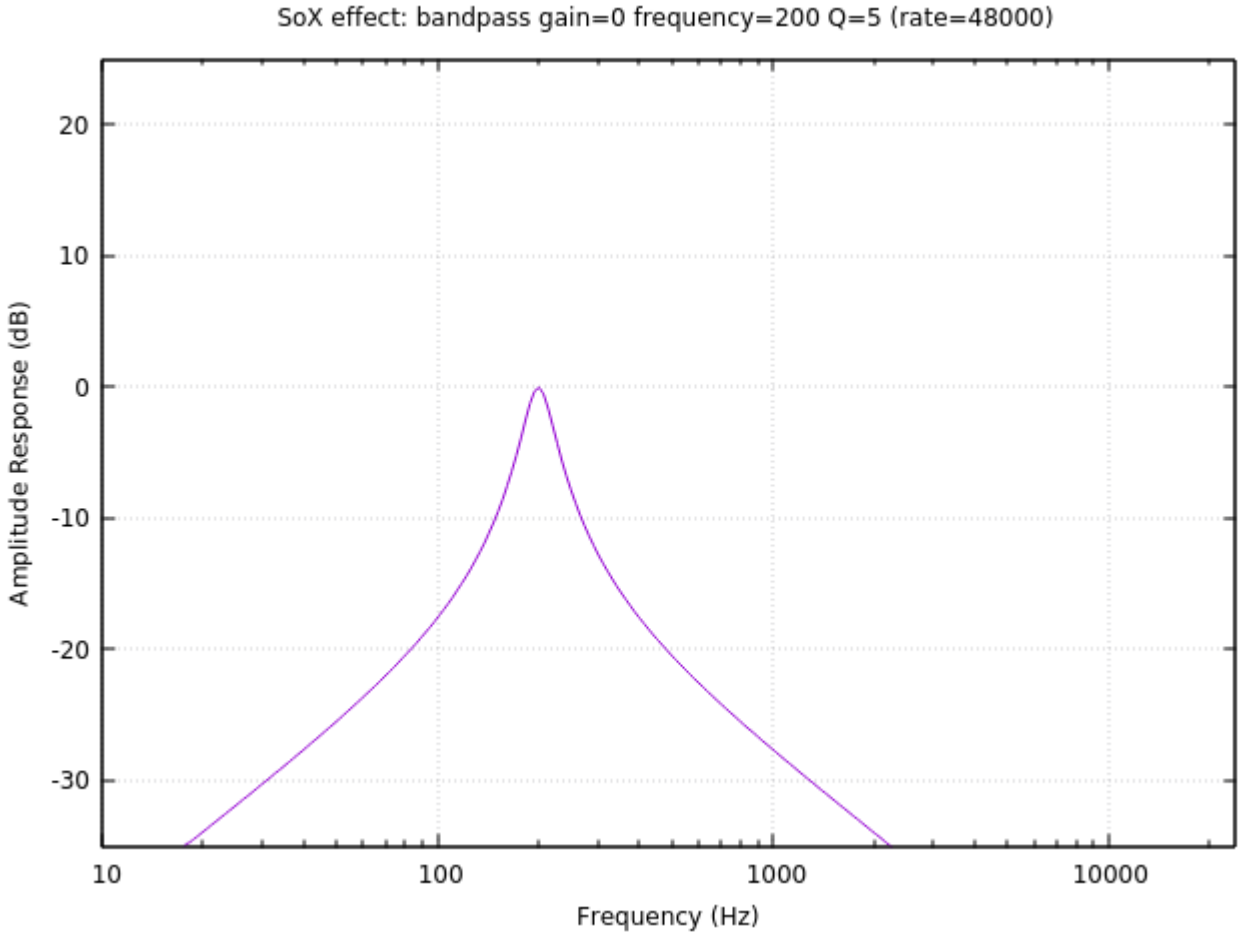
sox --plot gnuplot s6.wav -n lowpass -1 500 > 1lp.plt



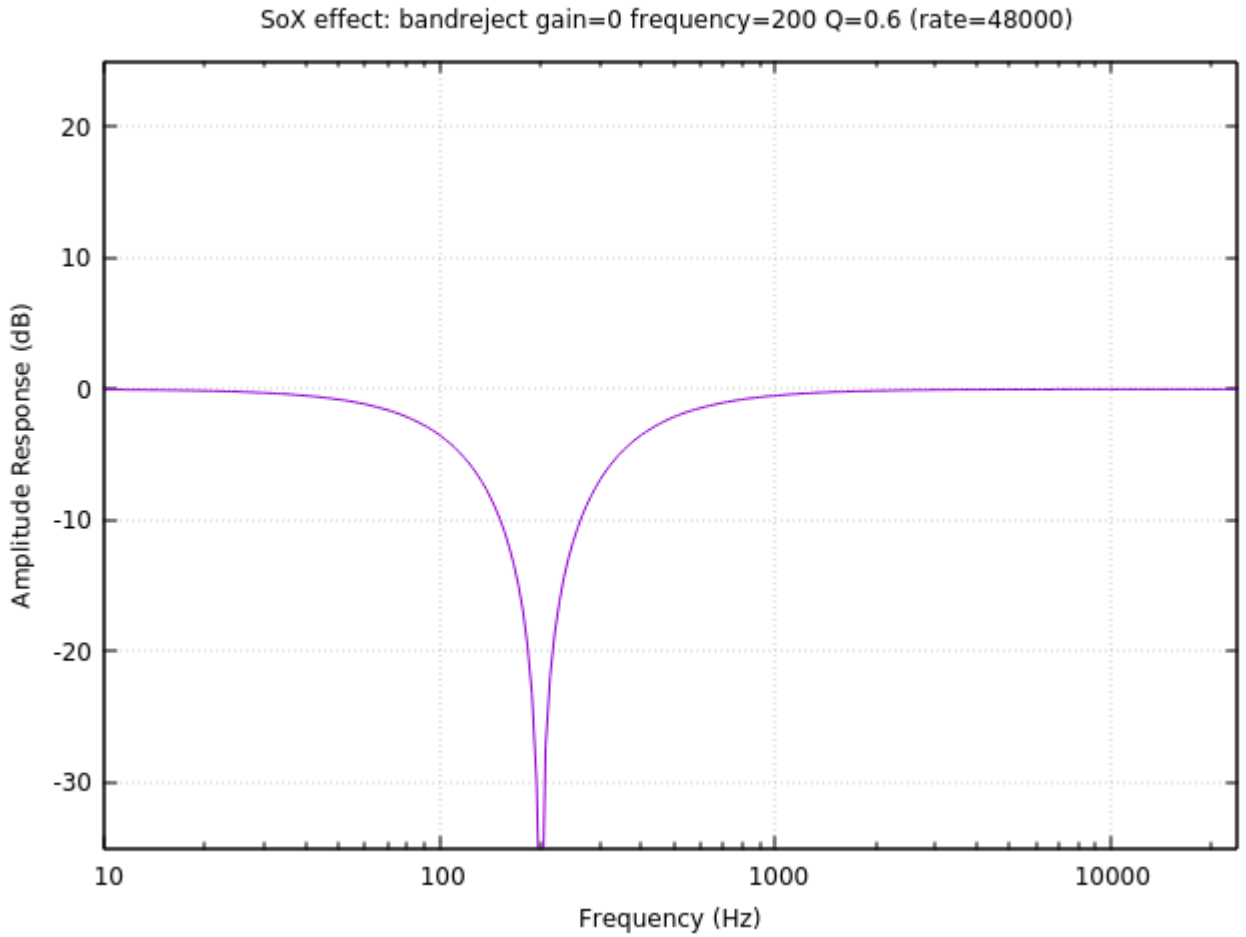
sox --plot gnuplot s6.wav -n highpass -2 2000 0.8q > 2hp.plt



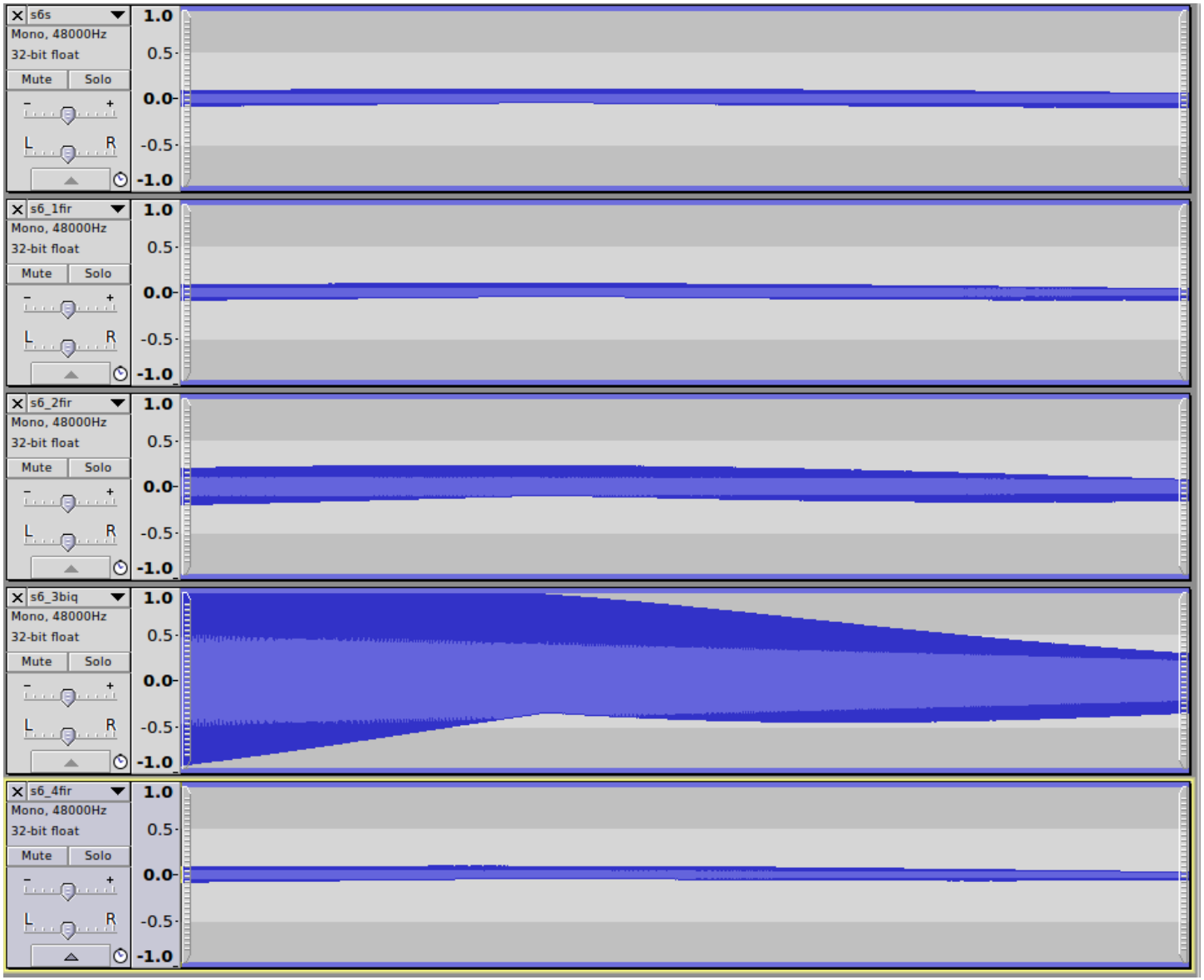
sox --plot gnuplot s6.wav -n bandpass 200 5q > 3bp.plt



sox --plot gnuplot s6.wav -n bandreject   200 0.6q > 4br.plt



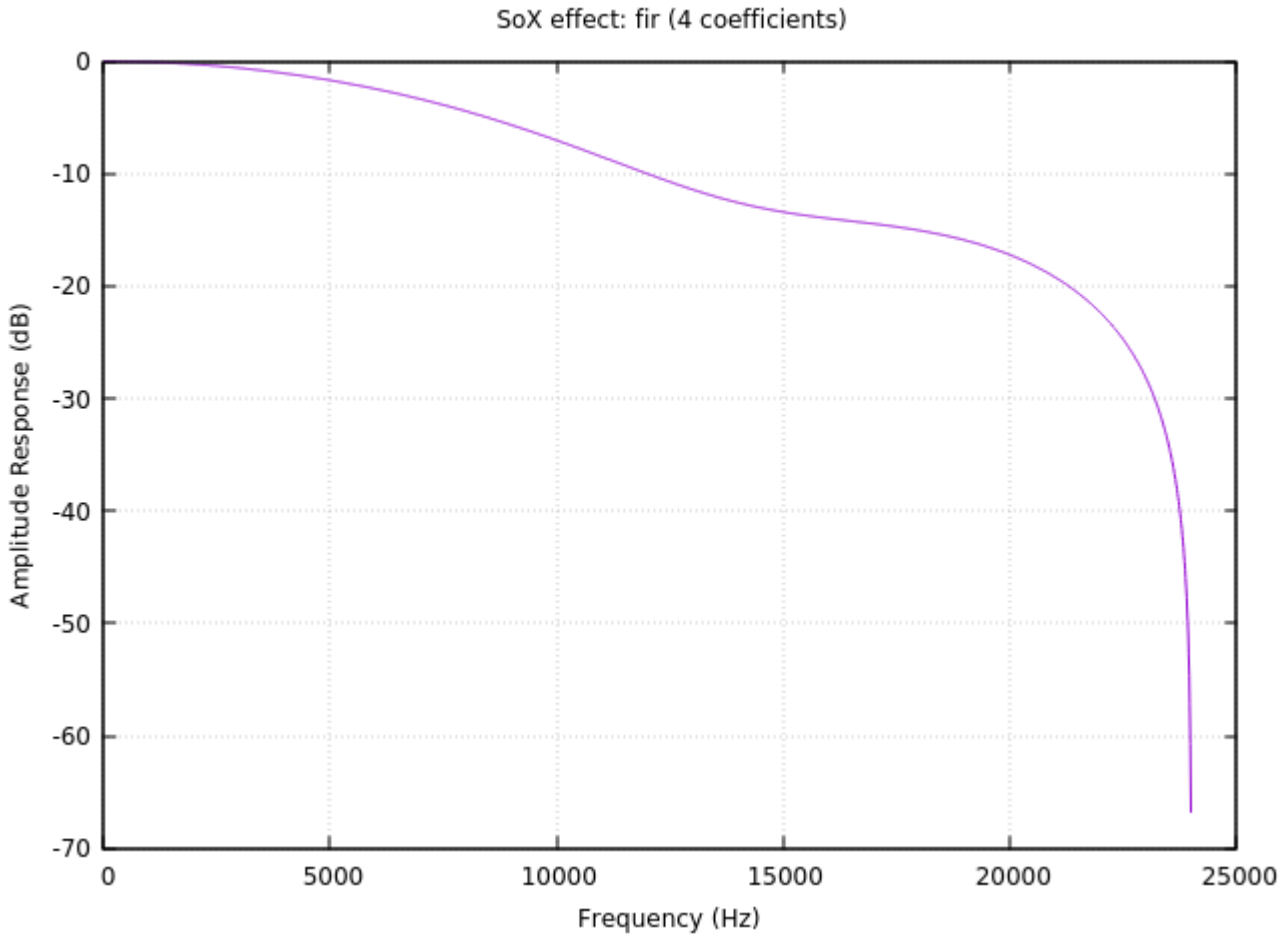
```
sox s6.wav s6s.wav  gain -20
sox s6s.wav s6_1fir.wav    fir 0.1 0.2 0.4 0.3
sox s6s.wav s6_2fir.wav    fir coeff.txt
sox s6s.wav s6_3biq.wav  biquad 0.6 0.2 0.4 1 -1.5 0.6
sox s6s.wav s6_4fir.wav    fir 0.2 0.2 0.2 0.2 0.2
```


s6s
Mono, 48000Hz
32-bit float
Mute
Solo
L
R
1.0
0.5
0.0
-0.5
-1.0
s6_1fir
Mono, 48000Hz
32-bit float
Mute
Solo
L
R
1.0
0.5
0.0
-0.5
-1.0
s6_2fir
Mono, 48000Hz
32-bit float
Mute
Solo
L
R
1.0
0.5
0.0
-0.5
-1.0
s6_3biq
Mono, 48000Hz
32-bit float
Mute
Solo
L
R
1.0
0.5
0.0
-0.5
-1.0
s6_4fir
Mono, 48000Hz
32-bit float
Mute
Solo
L
R
1.0
0.5
0.0
-0.5
-1.0

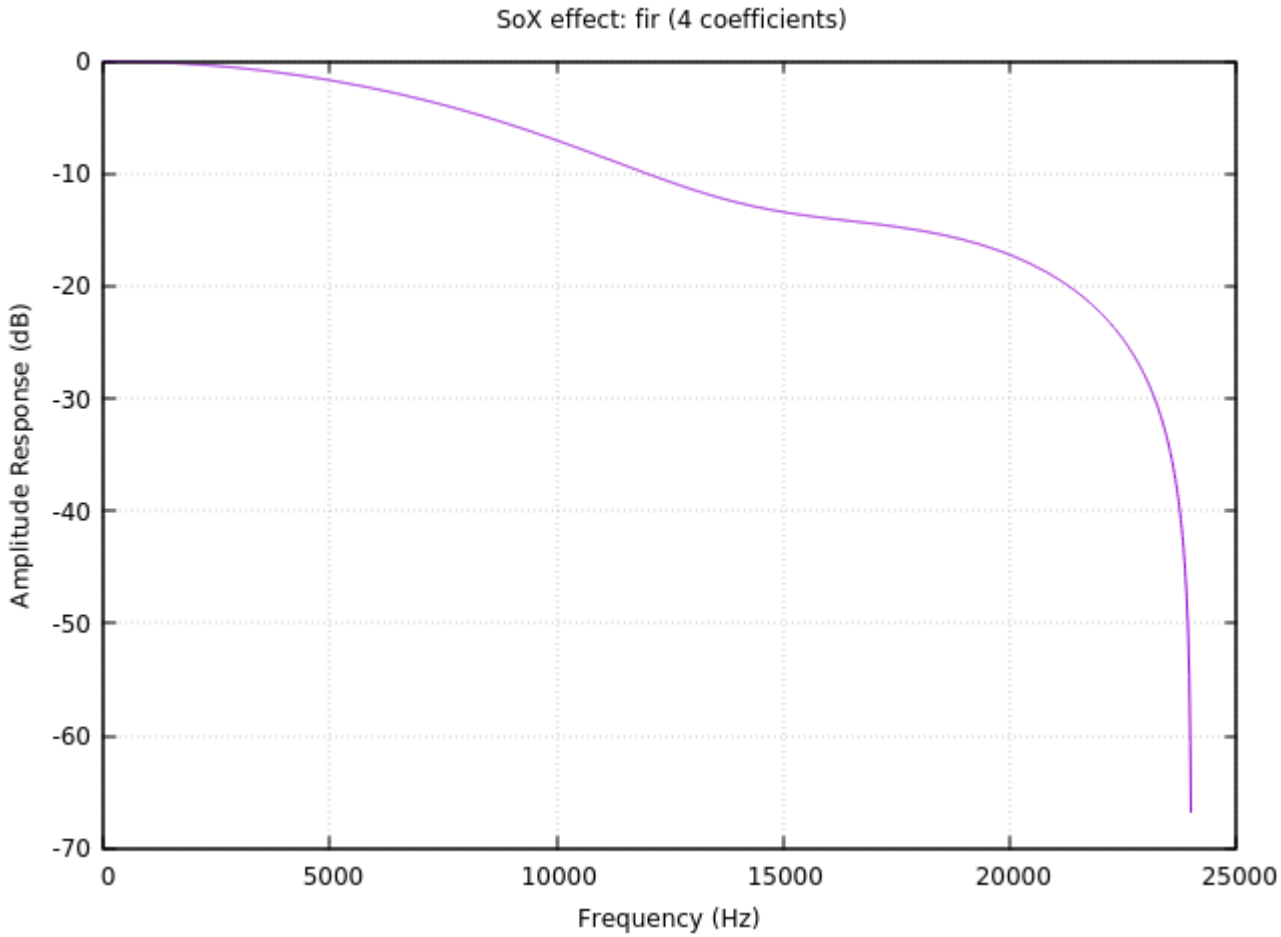
--plot gnuplot | octave

```
sox --plot gnuplot s6s.wav -n fir 0.1 0.2 0.4 0.3          >fir1.plt
sox --plot gnuplot s6s.wav -n fir coeff.txt                >fir2.plt
sox --plot gnuplot s6s.wav -n biquad .6 .2 .4 1 -1.5 .6 >fir3.plt
sox --plot gnuplot s6s.wav -n fir 0.2 0.2 0.2 0.2 0.2      >fir4.plt
```

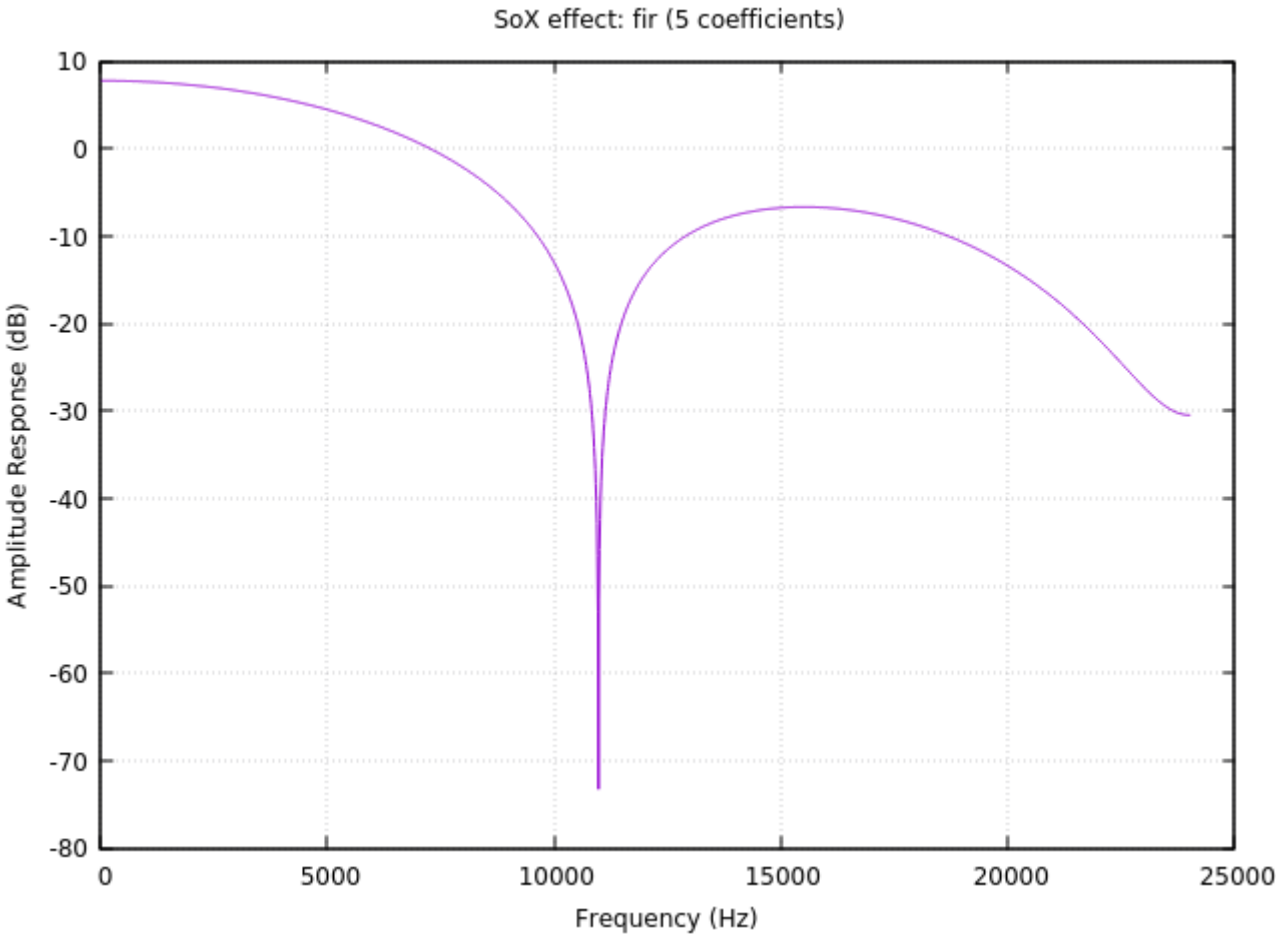
--plot gnuplot | octave



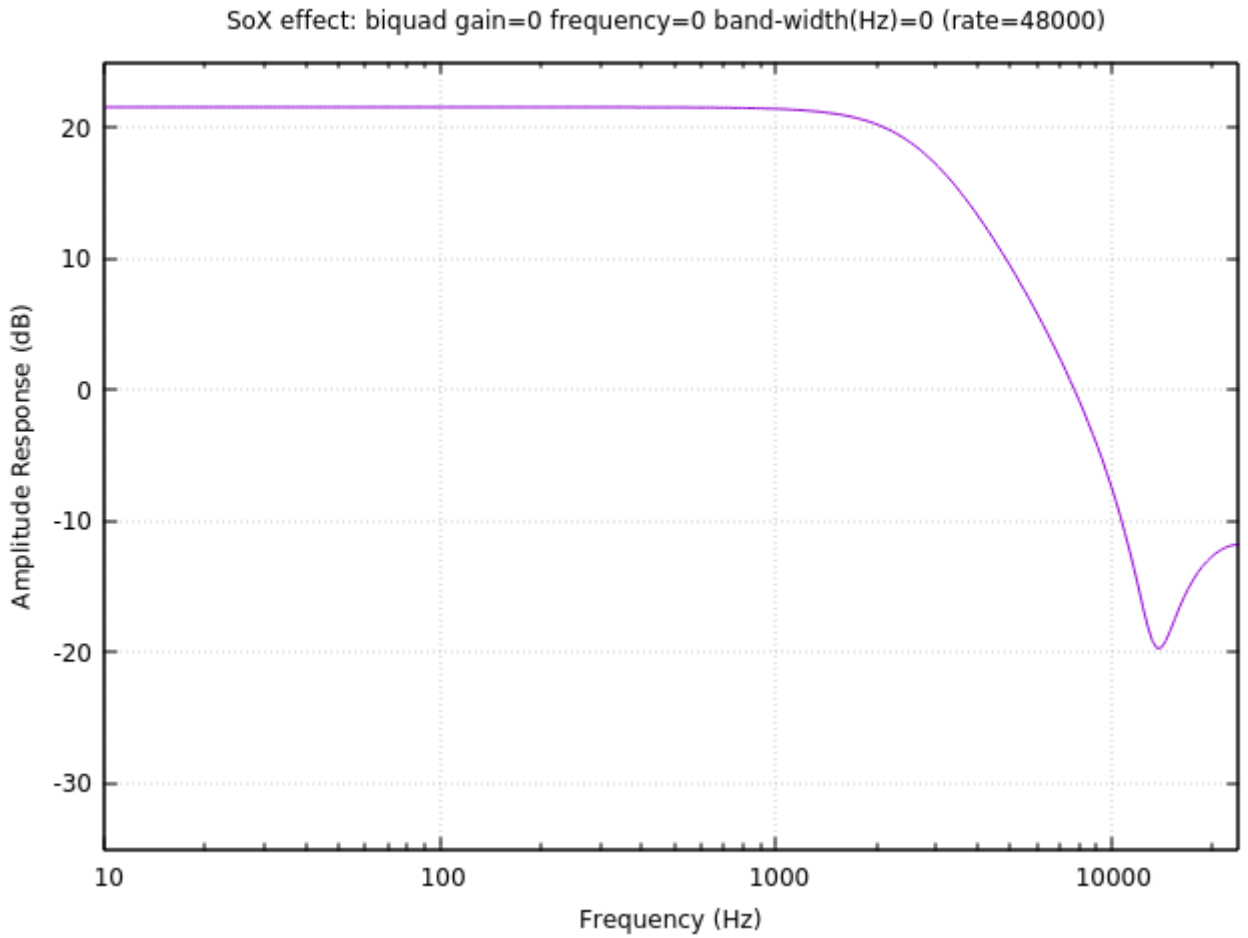
--plot gnuplot | octave



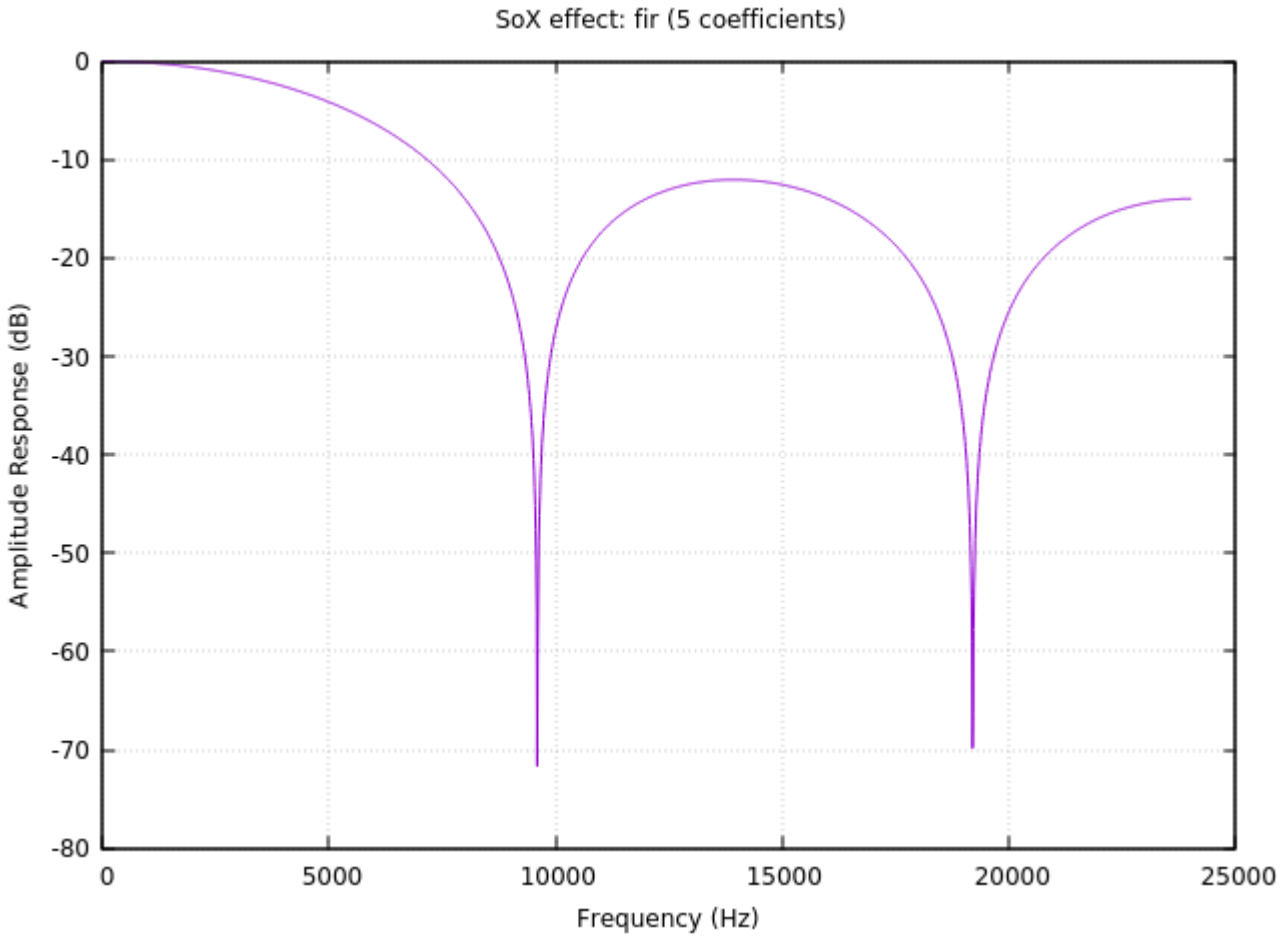
--plot gnuplot | octave



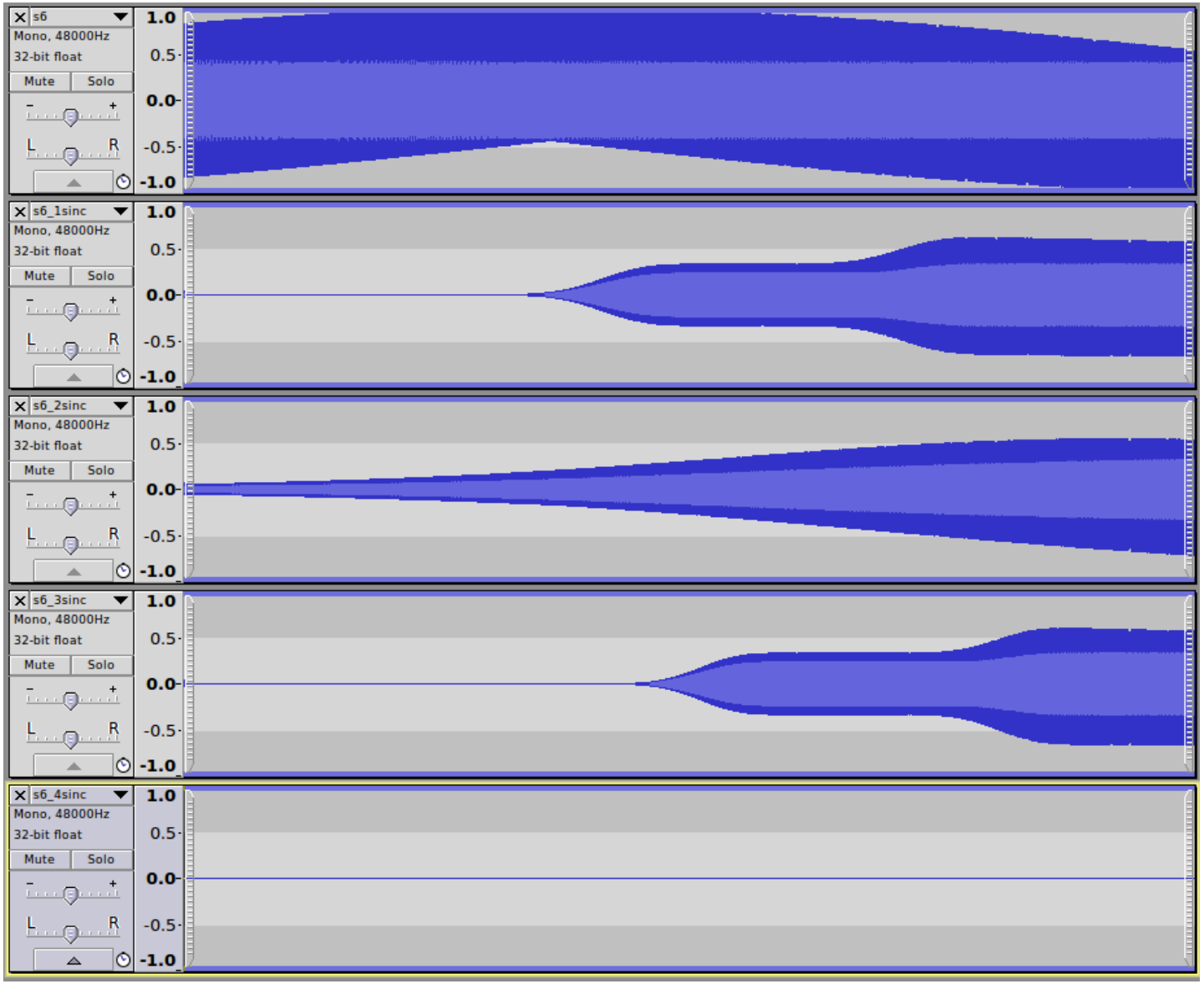
--plot gnuplot | octave



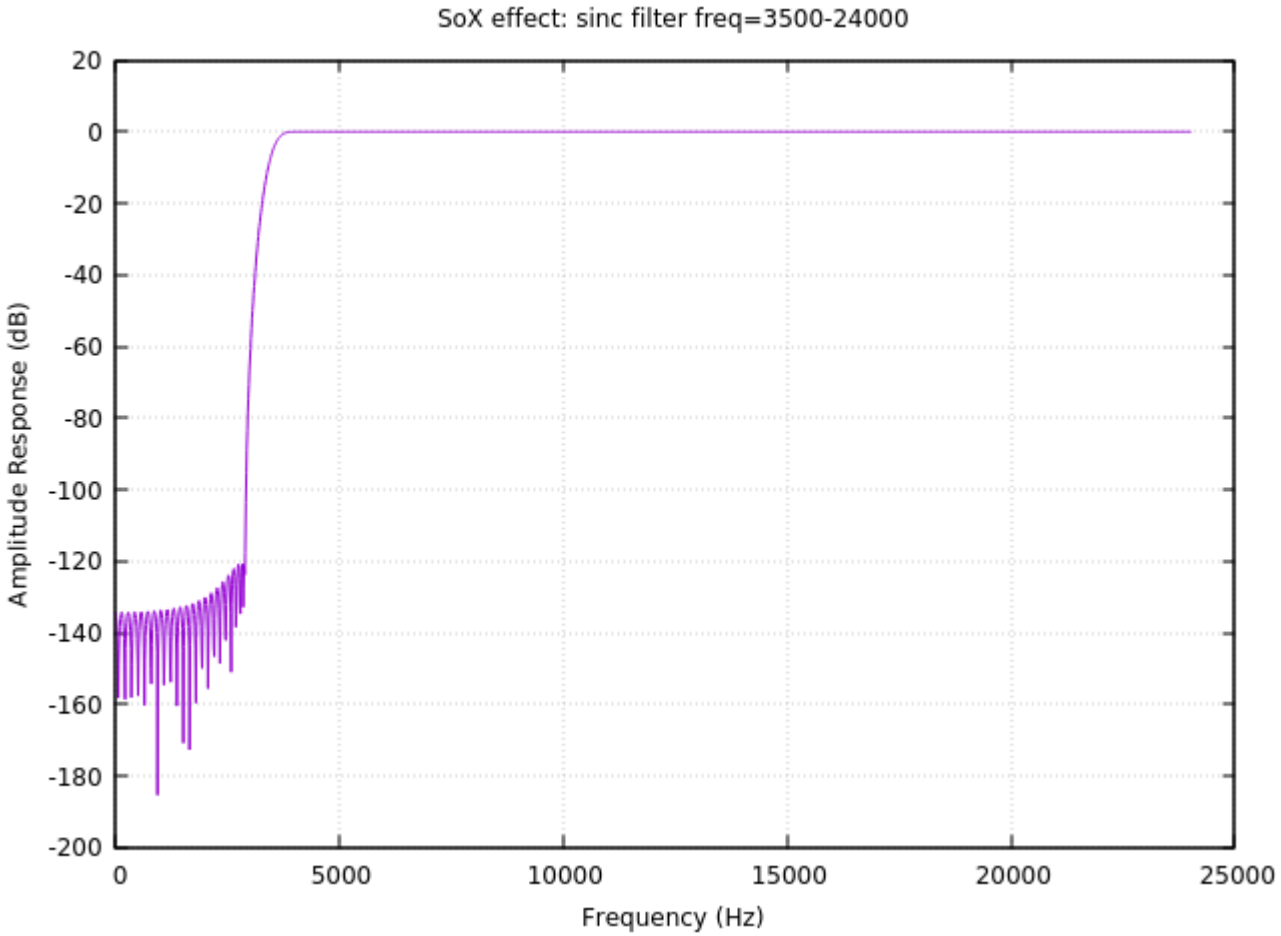
--plot gnuplot | octave

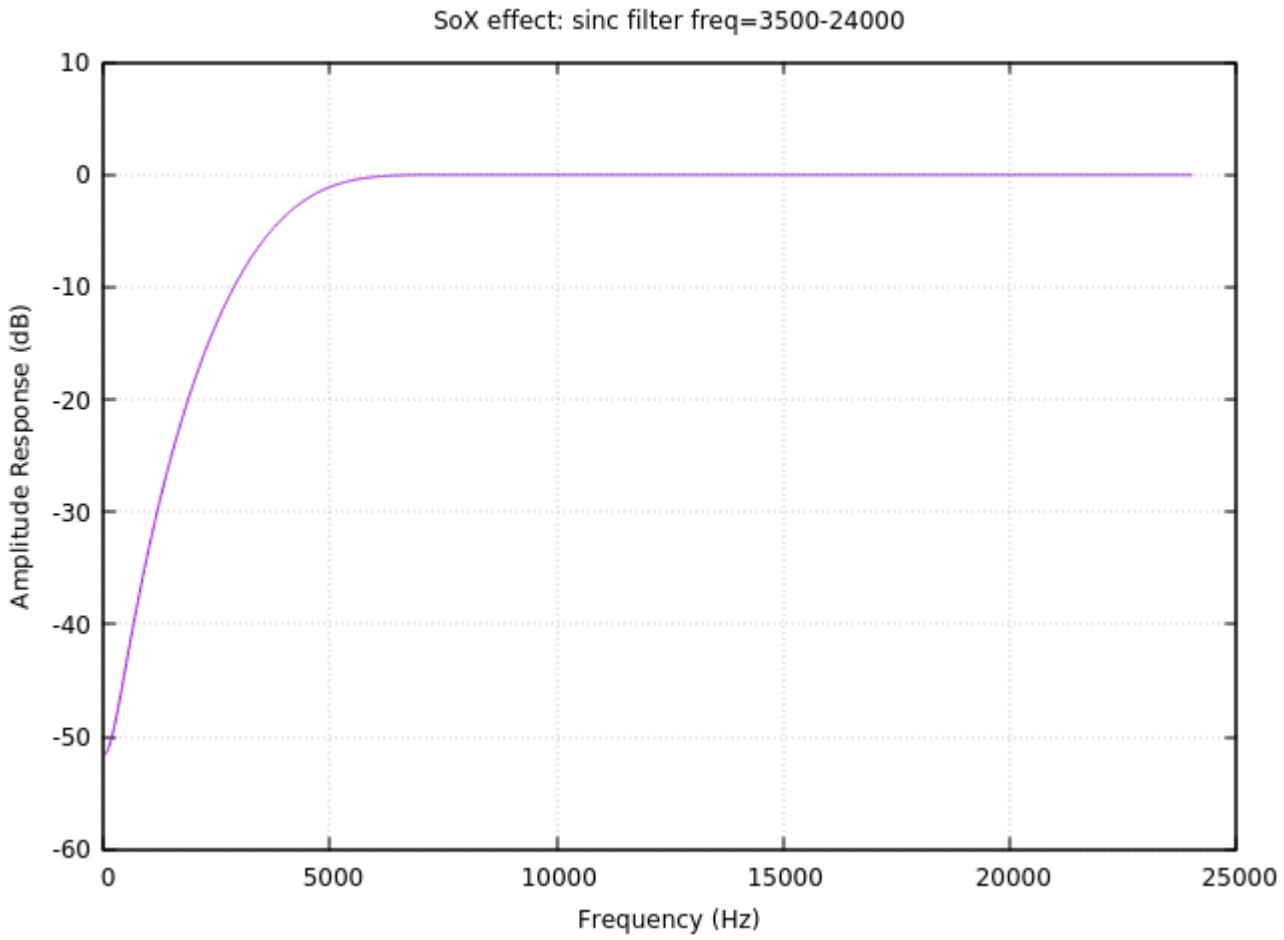


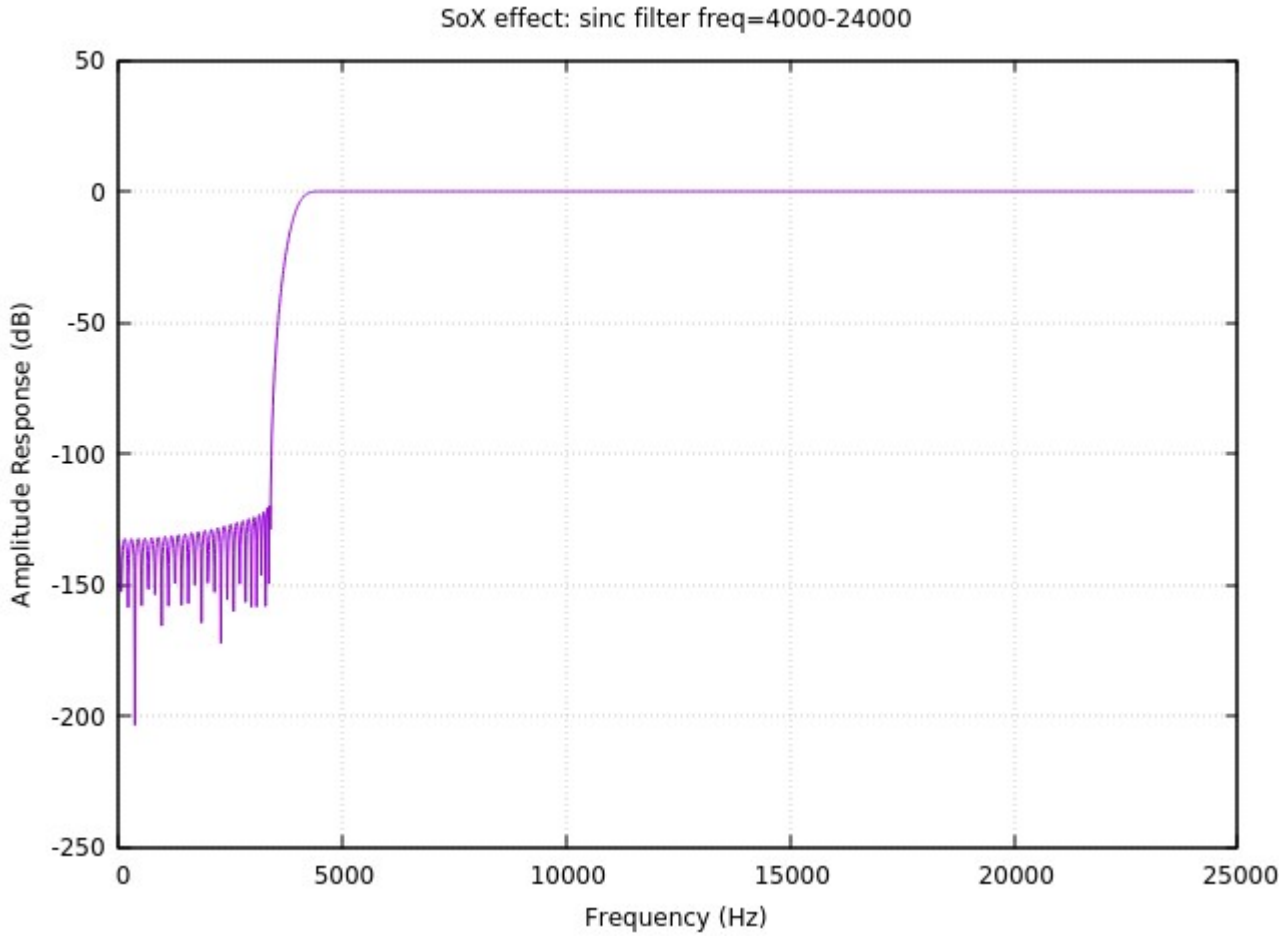
```
sox s6.wav s6_1sinc.wav sinc 3500
sox s6.wav s6_2sinc.wav sinc 3500 -n 43
sox s6.wav s6_3sinc.wav sinc 4000
sox s6.wav s6_4sinc.wav sinc 12000 -n 151
```

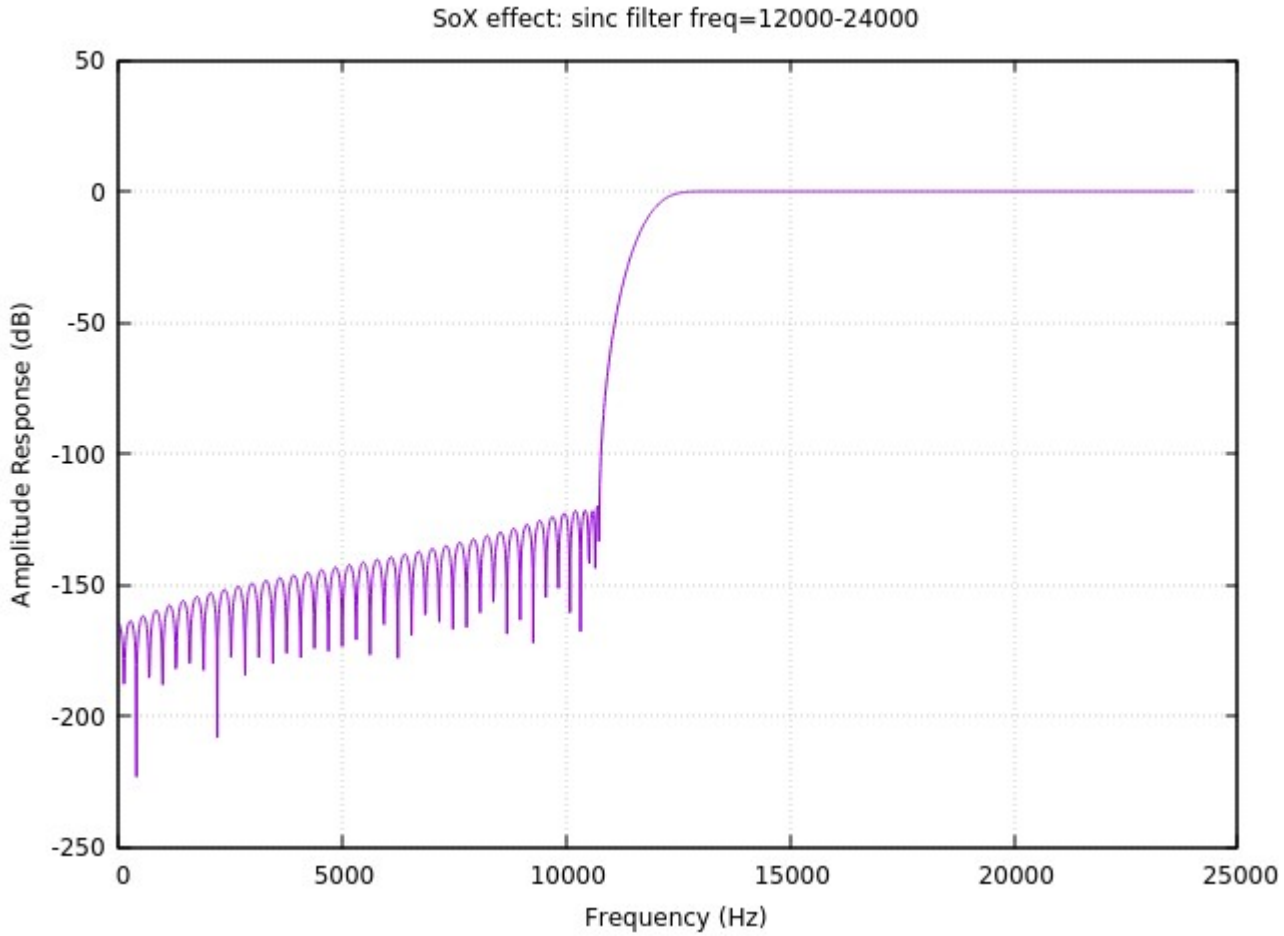

s6
Mono, 48000Hz
32-bit float
Mute
Solo
L
R
1.0
0.5
0.0
-0.5
-1.0
s6_1sinc
Mono, 48000Hz
32-bit float
Mute
Solo
L
R
1.0
0.5
0.0
-0.5
-1.0
s6_2sinc
Mono, 48000Hz
32-bit float
Mute
Solo
L
R
1.0
0.5
0.0
-0.5
-1.0
s6_3sinc
Mono, 48000Hz
32-bit float
Mute
Solo
L
R
1.0
0.5
0.0
-0.5
-1.0
s6_4sinc
Mono, 48000Hz
32-bit float
Mute
Solo
L
R
1.0
0.5
0.0
-0.5
-1.0

```
sox --plot gnuplot s6.wav -n sinc 3500              >sinc1.plt
sox --plot gnuplot s6.wav -n sinc 3500 -n 43        >sinc2.plt
sox --plot gnuplot s6.wav -n sinc 4000              >sinc3.plt
sox --plot gnuplot s6.wav -n sinc 12000 -n 151      >sinc4.plt
```


SoX effect: sinc filter freq=3500-24000
Amplitude Response (dB)
20
0
-20
-40
-60
-80
-100
-120
-140
-160
-180
-200
0
5000
10000
15000
20000
25000
Frequency (Hz)


SoX effect: sinc filter freq=3500-24000
Amplitude Response (dB)
10
0
-10
-20
-30
-40
-50
-60
0
5000
10000
15000
20000
25000
Frequency (Hz)


SoX effect: sinc filter freq=4000-24000
Amplitude Response (dB)
50
0
-50
-100
-150
-200
-250
0
5000
10000
15000
20000
25000
Frequency (Hz)


SoX effect: sinc filter freq=12000-24000
Amplitude Response (dB)
50
0
-50
-100
-150
-200
-250
0
5000
10000
15000
20000
25000
Frequency (Hz)

References

[1] F. Auger, Signal Processing with Free Software : Practical Experiments